**ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**

**KHOA KỸ THUẬT MÁY TÍNH**

**PHÙNG ĐÀO VĨNH CHUNG – 13521099**

**HUỲNH NHẬT QUANG - 13520673**

**KHÓA LUẬN TỐT NGHIỆP**

**HỆ THỐNG CHẨN ĐOÁN SỨC KHỎE**

**HEALTH CARE SYSTEM**

**KỸ SƯ NGÀNH KỸ THUẬT MÁY TÍNH**

**TP. HỒ CHÍ MINH, NĂM 2018**

**ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**

**KHOA KỸ THUẬT MÁY TÍNH**

**PHÙNG ĐÀO VĨNH CHUNG – 13521099**

**HUỲNH NHẬT QUANG – 13520673**

**KHÓA LUẬN TỐT NGHIỆP**

**HỆ THỐNG CHẨN ĐOÁN SỨC KHỎE**

**HEALTH CARE SYSTEM**

**KỸ SƯ NGÀNH KỸ THUẬT MÁY TÍNH**

**GIẢNG VIÊN HƯỚNG DẪN**

**TRẦN NGỌC ĐỨC**

**TP. HỒ CHÍ MINH, NĂM 2018**

**DANH SÁCH HỘI ĐỒNG BẢO VỆ KHÓA LUẬN**

Hội đồng chấm khóa luận tốt nghiệp, thành lập theo Quyết định số 75/QĐ-ĐHCNTT ngày 24/1/2018 của Hiệu trưởng Trường Đại học Công nghệ Thông tin.

* 1. TS. Trịnh Lê Huy – Chủ tịch.
  2. Ths. Phan Đình Duy – Thư ký.
  3. Ths. Lê Hoài Nghĩa – Ủy viên.

**ACKNOWLEDGEMENTS**

In order to complete this graduation thesis, we would like to express our sincere thanks to the teachers of University of Information Technology in general and the Faculty of Computer Engineering in particular for imparting knowledge and valuable experience during the past 5 years to us.

We would like to send our sincere thanks to Mr. Tran Ngoc Duc, who devoted his precious time directed us to complete the graduation thesis.

We deeply thanks to the help of our friends, seniors who helped us to find information during the course.

Once again, we would like to thank all those who are interested in our thesis. However, in the process of working there may be some unavoidable mistakes, we are looking forward to receiving comment of this thesis..

Phùng Đào Vĩnh Chung – Huỳnh Nhật Quang

# 

|  |  |
| --- | --- |
| ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH  **TRƯỜNG ĐẠI HỌC**  **CÔNG NGHỆ THÔNG TIN** | **CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM**  **Độc Lập - Tự Do - Hạnh Phúc** |

**ĐỀ CƯƠNG CHI TIẾT**

|  |  |
| --- | --- |
| **TÊN ĐỀ TÀI: HỆ THỐNG CHẨN ĐOÁN SỨC KHỎE** | |
| **Cán bộ hướng dẫn: Ths. Trần Ngọc Đức** | |
| **Thời gian thực hiện:**Từ ngày 01/09/2017 đến ngày 17/01/2018 | |
| **Sinh viên thực hiện:**  **Phùng Đào Vĩnh Chung – 13521099**  **Huỳnh Nhật Quang – 13520673** | |
| **Nội dung đề tài:**  + **Bài toán nghiên cứu**: Nâng cấp và phát triển hệ thống chẩn đoán sức khỏe từ khóa luận  “Thiết kế hệ thống phần cứng đo các chỉ số cơ thể tích hợp lưu trữ thông tin qua mạng”:   * Bổ xung các tính năng mới * Tăng độ chinh xác cho hệ thống * Phát triển ứng dụng di động hỗ trợ người dùng theo dõi tính trạng sức khỏe   + **Mục tiêu**:   * Xây dựng hệ thống sử dụng các module hoạt động theo dạng module * Tăng độ chính xác trong việc đo các chỉ số sức khỏe * Phát triển ứng dụng di động giúp theo dõi tình trạng sức khỏe của người dùng, tình trạng vận động, mức độ calories tiêu thụ, khả năng bị bệnh tim   + **Đối tượng và phạm vi nghiên cứu**: Tập trung vào đối tượng là người Việt Nam ở độ tuổi thiếu niên và trưởng thành. Nghiên cứu tập trung vào việc xây dựng một hệ thống gồm những module độc lập, hoạt động theo dạng plug-and-play, tăng độ chính xác cho các module, tạo một ứng dụng điện thoại hỗ trợ cho việc theo dõi sức khỏe và chẩn đoán bệnh cho người dùng.  + **Phương pháp nghiên cứu**:   * Đo những chỉ số sinh học cơ bản của người sử dụng (nhịp tim, nồng độ oxy, chiều cao, cân nặng, nhiệt độ cơ thể) bằng hệ thống cảm biến * Thu thập dữ liệu từ những cảm biến có sẵn trên điện thoại (đo thời gian sử dụng smartphone, kiểm tra tư thế ngồi) và thông tin người dùng nhập * Dùng thông tin thu được, ứng dụng đưa ra cảnh báo tới người dùng thông qua các rules đã được lập trình sẵn   + **Kết quả mong đợi**   * Hệ thống có khả năng hoạt động tự động với các module hoạt động theo dạng module * Kết quả đo đạc các chỉ số sức khỏe của người dùng đạt độ chính xác ở mức có thể chấp nhận được (sai số ± 5%) * Ứng dụng di động có thể cập nhật thông tin về trạng thái sức khỏe của từng người dùng, tính lượng calories người dùng tiêu thụ và chẩn đoán khả năng bị bệnh tim của người dùng | |
| **Kế hoạch thực hiện:** Quá trình nghiên cứu được thực hiện gồm 3 giai đoạn chinh:  **+ Giai đoạn 1 (9-10/2017 ):**   * Nghiên cứu về cách hoạt động của module * Nghiên cứu về cách hoạt động của hệ thống cũ của luận văn trước   + **Giai đoạn 2 (10 – 12/2017):**   * Tiến hành tạo nguyên mẫu của các module * Tạo server cho hệ thống * Phát triển ứng dụng trên điện thoại   **+ Giai đoạn 3 (12/2017 – 1/2018):**   * Tối ưu hóa hệ thống * Tiến hành Unit Test, Functional Test * Viết báo cáo quá trình và kết quả nghiên cứu | |
| **Xác nhận của CBHD**  (Ký tên và ghi rõ họ tên)  Ths. Trần Ngọc Đức | **TP. HCM, ngày 17 tháng 1 năm 2018**  **Sinh viên**  (Ký tên và ghi rõ họ tên)  Phùng Đào Vĩnh Chung  Huỳnh Nhật Quang |

**ABREVIATION LIST**

|  |  |
| --- | --- |
| DC | Direct Curent |
| I2C | Inter-Integrated Circuit |
| IDE | Integrated Development Environment |
| UART | Universal asynchronous receiver/transmitter |
| MCU | Micro Control Unit |
| Spo2 | Pulse Oximeter |
| PWM | Pulse-Width Modulation |
| RAM | Random Access Memory |
| ROM | Read Only Memory |

# MENU

[ABSTRACT 1](#_Toc508658551)

[Chapter 1. THESIS OVERVIEW 2](#_Toc508658552)

[1.1. Background 2](#_Toc508658553)

[1.2. Thesis overview 2](#_Toc508658554)

[1.3. Problem statement 2](#_Toc508658555)

[1.4. Purpose, target and scope 3](#_Toc508658556)

[1.4.1. Purpose of this thesis 3](#_Toc508658557)

[1.4.2. Scope 4](#_Toc508658558)

[1.5. Theoretical and practical meaning of the topic 4](#_Toc508658559)

[1.6. Advantage and Disadvantage 5](#_Toc508658560)

[1.6.1. Advantage 5](#_Toc508658561)

[1.6.2. Disadvantage 5](#_Toc508658562)

[Chapter 2. THEORY AND EPERIMENTAL STUDY 6](#_Toc508658563)

[2.1. Theory 6](#_Toc508658564)

[2.1.1. Survival signs 6](#_Toc508658565)

[2.1.2. Body temperature 6](#_Toc508658566)

[2.1.3. Heart rate 8](#_Toc508658567)

[2.1.4. Blood oxygen levels 10](#_Toc508658568)

[2.1.5. Machine learning 12](#_Toc508658569)

[2.1.6. Support Vector Machine 13](#_Toc508658570)

[2.2. Experiamental study 15](#_Toc508658571)

[2.2.1. System overview 15](#_Toc508658572)

[2.2.2. Devices and tools overview: 15](#_Toc508658573)

[2.2.3. Study board Raspberry Pi 3 15](#_Toc508658574)

[2.2.4. Arduino Uno overview 17](#_Toc508658575)

[2.2.5. Temperature sensor overview: 18](#_Toc508658576)

[2.2.6. Heart rate and spo2 sensor overview 19](#_Toc508658577)

[2.2.7. Stm32f103c8t6 overview: 20](#_Toc508658578)

[2.2.8. Heart disease prediction: 21](#_Toc508658579)

[2.2.9. Android app overview 22](#_Toc508658580)

[2.2.10. Operating system 24](#_Toc508658581)

[2.2.11. Connection between Raspberry Pi and modules 25](#_Toc508658582)

[2.2.12. Communicate with oxygen in blood measuring device 26](#_Toc508658583)

[2.2.13. H-Bridge circuit: 26](#_Toc508658584)

[2.2.14. Encoder: 29](#_Toc508658585)

[2.2.15. Ultra sonic sensor: 31](#_Toc508658586)

[2.2.16. USART overview 32](#_Toc508658587)

[2.2.17. I2C overview 32](#_Toc508658588)

[2.2.18. Arduino IDE: 34](#_Toc508658589)

[Chapter 3. SYSTEM ANALYSIS AND DESIGN 35](#_Toc508658590)

[3.1. System general model 35](#_Toc508658591)

[3.1.1. Connection between modules and boards 35](#_Toc508658592)

[3.1.2. System operation process: 36](#_Toc508658593)

[3.1.3. Module operation process: 37](#_Toc508658594)

[3.1.4. Read data from Heart-Rate and Pulse-Oximetry sensor: 38](#_Toc508658595)

[3.1.5. Read data from temperature sensor: 40](#_Toc508658596)

[3.1.6. Android app working process overview 41](#_Toc508658597)

[3.1.7. Communication between Raspberry Pi and Server: 41](#_Toc508658598)

[3.1.8. Heart disease prediction 42](#_Toc508658599)

[Chapter 4. TEST RESULTS AND EVALUATION 43](#_Toc508658600)

[4.1. Test process 43](#_Toc508658601)

[4.1.1. Components for testing 43](#_Toc508658602)

[4.1.2. Steps to use the system: 45](#_Toc508658603)

[4.1.3. Result: 46](#_Toc508658604)

[Chapter 5. CONCLUSION AND REQUEST 59](#_Toc508658605)

[5.1. Conclusion: 59](#_Toc508658606)

[5.1.1. Achievements: 59](#_Toc508658607)

[5.1.2. Failure: 60](#_Toc508658608)

[5.2. Development in the future 60](#_Toc508658609)

# FIGURE MENU

Figure 2-1: System overview diagram 12

Figure 2-2: Raspberry Pi 3 13

Figure 2-3: Arduino Uno 14

Figure 2-4 Infrared sensor MLX 90614 15

Figure 2-5 Heart-Rate and Pulse-Oximetry Monitor 16

Figure 2-6 STM32F103C8T6 18

Figure 2-7 App overview diagram 19

Figure 2-8 A sketch of a typical Mbed board's hardware architecture 21

Figure 2‑9 H-Bridge circuit 25

Figure 2-10 Operational principles 25

Figure 2-11 Encoder 27

Figure 2-12 Channel A and B on Encoder 28

Figure 2-13 Ultra sonic sensor 29

Figure 2-14 I2C diagram 31

Figure 3-1: Connection diagram 33

Figure 3-2: Operation process 34

Figure 3-3: Operation process 35

Figure 3-4: Module general diagram 36

Figure 3-5: Spo2 Sensor Schematic 36

Figure 3-6: Spo2 and HR flowchart 37

Figure 3-7: Schematic Sensor MLX90615 38

Figure 3-8: Schematic Sensor MLX90615 38

Figure 3-9: Data upload to Server diagram 39

Figure 4-1: Download code into Arduino Uno 40

Figure 4-2: Download code into Stm32f103c8t6 41

Figure 4-3: Components after connected 41

Figure 4-4: system process diagram 42

# TABLE MENU

Table 2.1: Classification of physical condition according to body temperature 8

Table 2.2: Classify heart rate condition in human 10

Table 2.3 Classification of oxygen levels in human blood 11

Table 4.1 Temperature result 43

Table 4.2 Spo2 result 45

Table 4.3 Heart rate result 46

Table 4.4 Weight result 49

Table 4.5 Height result 51

Table 4.6 Result from Server 53

# ABSTRACT

The issue of health care for people is always takes a big role in our society. Monitoring health status is one of the most important and simplest method to take care of our health. In order to meet that need, our team has researched and implemented this project.

The thesis focused on researching and upgrading the previous system, turning body temperature, heart rate and spo2 sensors into plug-and-play devices with the central controller unit is Raspberry Pi. The system links to phone application through the cloud server to monitor user’s health status as well as movement status to measure the burned calories to provide predictions, warnings about the health situation. The thesis report is splitted into 5 chapters:

* **Chapter 1.** Thesis overview.
* **Chapter 2.** Theory and experimental study
* **Chapter 3.** System analysis and design
* **Chapter 4.** Test result and evaluation.
* **Chapter 5.** Conclusion and request

# THESIS OVERVIEW

## Background

According to a report by the Ministry of Health and the World Health Organization (WHO), the mortality rate from non-communicable diseases in Vietnam is 73% and tends to increase, causing a great implication for the society and the economy of the country. And the majority of it is disease related to cardiovascular, cancer, chronic respiratory diseases and diabetes. However, the health care system is still having many inadequacies.[12]

Monitoring health status is one of the simplest but most useful methods to take care people health. The health status can be assessed through vital signs. Vital sign is the sign that reflect the physiological functions of the body. Monitoring vital signs can help detect abnormalities within the body which provide predictable as well as timely intervention. This can contribute to improve the health of the user and limit the unwanted consequences from illness.

## Thesis overview

The issue of health care for people is always takes a big role in our society. Monitoring health status is one of the most important and simplest method to take care of our health. In order to meet that need, our team has researched and implemented this project.

The thesis focused on constructing a body temperature measurement, heart rate, and oxygen levels in the blood through the sensor, a measurement device with the central processing unit Raspberry Pi. Can be linked to phone application through the cloud server to monitor user’s health status as well as movement status using machine learning to measure the number of calories they consume to provide predictions, warnings about the health situation.

## Problem statement

Nowadays, devices that measure basic body parameters such as temperature, blood pressure, heart rate and oxygen levels in the blood become so familiar in human life that it is indispensable for diagnosis, prevention and treatment of diseases.

There are a number of studies in the world involving instruments for measuring human health indicators. Devices that measure body temperature, heart rate, blood pressure, and blood oxygen levels are usually only capable of measuring one or two health indicators, while those with integrated capabilities measure multiple health indicators usually very expensive. Among the most popular health indicators today are Keito's K6, K7, ... Keito, Altura Multi from Health Kiosk or Millenium from Healthcheck Services, These products are capable of measuring the health indicators such as weight, height, blood pressure, heart rate, and body fat.

In Vietnam there are also topics related to health care equipment. Most of the domestic topics are measured by a specific indicator and are not able to communicate over the network. And most of them don’t have the plug-and-play feature or an application goes with them for users to track their health. Meanwhile, the medical devices currently used in the country are mostly imported products with high cost and maintenance difficulties.

One of the disadvantages of current health meter devices is that they usually only display results to the user immediately after the measurement, not yet capable of storing previous measurement results as well as no system part. Soft support for display, storage, observation of measurement results as well as giving health recommendations to users

## Purpose, target and scope

### Purpose of this thesis

Our thesis focused on researching and upgrading the previous system to be able to measure body mass indexes such as temperature, oxygen levels in blood, heart rate that can be bound to the software via an intermediate server. Based on the indicators have been obtained combined with the information gets from the user and the phone's sensor system gives the prediction of the user heart rate and calorie intake so that the user can has a more accurate judgments on their state of health.

Details:

* Research central processing board: Raspberry Pi 3
* Research sensors and measuring devices: temperature, oxygen concentration, heart rate, height, weight
* Research about using related equipment: motor, distance measuring sensor, capacitive touch button, ...
* Learn about the communication standards UART and I2C to communicate between measuring devices and the center board
* Study Plug and Play mechanism so that each device is a standalone module, plugged in and powered by the Plug and Play.
* Study nodejs language to build server computer, java language in Android Studio environment to write software
* Research on deep learning to build a Heart Disease Prediction Model, showing the amount of calories people have consumed based on their activities.

### Scope

Our team focuses on investigating and designing a gadget that has body-measuring devices work as separated modules, and connecting by using plug-and-play mechanism. Data will be stored on the server and processed before displayed on mobile app.

Within the research scope and design of this instrumentation. We focus on Vietnamese teenagers and adults.

Since we can’t afford experiment a real foreigner so they are not in our research area.

## Theoretical and practical meaning of the topic

To design and carry out this instrumentation have a big influence on fasting and comprehensive monitoring health status for patients who visits medical facilities in particular and the Vietnamese people in general. When the health monitoring equipment in Vietnam have not really developed yet.

This device will save a lot of time for medical examination. In addition, placing this device in commercial buildings is also extremely convenient for people to quickly check their health status..

## Advantage and Disadvantage

### Advantage

Receive the enthusiastic help and valuable comments from the thesis instructor. Beside we also have received a lot of support from the Faculty of Computer Engineering, such as the loan equipment, research tools, working rooms and thanks to teachers who help us in finding and supplying a lot of useful materials as well as suggestions to help us complete the thesis.

Thanks to the members in our team who have collaborated on this project with a good assignment plan and sense of responsibility contributed to the success of the thesis. Moreover, the support from friends, relatives, especially the knowledge and experience shared by other research groups is also a great source helping us complete this thesis.

### Disadvantage

Although we have been trying very hard to complete this thesis but it is inevitable difficulties due to the lack of experience and knowledge: difficulty in optimizing the system so that the timing is optimized without reducing the accuracy of each measuring device; Add the deep learning to the software to predict heart disease, calorie intakes based on the predictions of activity that need to be tested more to increase the accuracy to the highest; difficulties in finding the right accessories and tools..

# THEORY AND EPERIMENTAL STUDY

## Theory

### Survival signs

Signs of survival include four basic signs: body temperature, blood pressure, heart rate and breathing [2][10]. Signs of survival reflect the physiological functions of the body. On this day beside the four vital signs of life, oxygen levels in the blood are being considered as the fifth vital sign.

Monitoring of vital signs helps detect abnormal changes within the body such as cardiovascular, respiratory, neurological, and endocrine systems in the body. In addition, changes in the physiological state of the body, physical, environmental and psychological responses affect these signals. These changes can happen very suddenly or for a long time. Therefore, any abnormal changes of vital signs need to be recorded for timely intervention.

Monitoring of vital signs for various purposes such as periodic health checks, disease monitoring, disease progression, diagnosis, follow-up of care results, complication detection disease, concludes the survival of the sick.

### Body temperature

#### **Human body temperature overview**

Humans are homoeothermic, body temperature usually around 37 oC even when interact with different temperature.

An average person can be exposed to temperatures ranging from less than 12.8 oC to 54.5 oC in low humid environment and body temperature is still raging around 36.1 oC – 37.8 oC. .

When the body temperature reaches 40oC - 41oC, humans can charge in just a short time. When the body temperature is 42oC or higher, there is a rapid degradation of the proteins in the cell and leads to death. All the cellular, biochemical and enzymatic reactions depend on temperature; therefore, regulating correct body temperature is necessary for humans.

#### **Body temperature types**

Body temperature is a result of Thermogenesis and Exothermic. Depending on the temperature measurement location, people classified it into 2 types Core temperature and body surface temperature.

Core temperature:

- Core temperature is the temperature of a body specifically in deep structures of the body such as the liver, brain and other viscera, as known as core body temperature. Average core temperature is usually fluctuate around 36 oC – 37.5 oC but mostly about 36.5 oC – 37 oC.

Human body surface temperature

* Human body surface temperature is the temperature of the skin and the organization under the skin. The temperature depends on body parts and environmental temperature. In the room temperature (24 oC – 25 oC), temperature of the head is 35 oC, of the palm area is 32.5 oC, of the arm is 31 oC; of hand and feet area is 29 oC.

#### **Factors affecting body temperature**

Human body temperature is influenced by many different factors such as:

* Locomotion:

Physical activities can increase core temperature up to 2oC or more. Core temperature can reach 38.5oC – 40oC when doing heavy physical activities, and can go up to 41oC when you have been overworked for too long.

* Biorhythm:

Body temperature decreased to lowest at night while you are asleep and increased slightly in the early morning. The body temperature reached maximum in the afternoon. The daily temperature variation is about 1° C.

* Age:

Kid’s body temperature usually higher than adult due to the increased of physical activities and metabolism. Premature babies, regular babies and elders body temperature are not very stable.

* Illness:

Hyperthermia can be seen in infections, hyperthyroidism or adrenal tumors can also be found in cholera.

It can be said that body temperature is a range of temperature around 36.8° C and depends a lot on other factors. When the body temperature goes higher or lower than this range, your body is in danger.

|  |  |
| --- | --- |
| **Classification of physical condition according to body temperature** | |
| Hypothermia | < 35.0oC |
| Normal | 36.5oC – 37.5oC |
| Hyperthyroidism | > 37.5oC – 38.3oC |
| High fever | > 40.0oC – 41.5oC |

Table 2.1 Classification of physical condition according to body temperature

### Heart rate

#### **Heart rate overview**

Heart rate is the number of contractions of the heart per minute. Heart rate depends on age, body weight, activity status (such as rest or active), diseases, medications.[14] Heart rate can also be affected by temperature. The most obvious factor is emotion. When we are stimulated, frightened, joyful or anxious our heart rate increases.

But these factors are mediated to bring the heart rate to stable state thanks to the rhythmic coordination of the central nervous system, cardiovascular system and chemical intermediates. Normal heart rate of an adult is around 60 – 100 beats/min.

#### **Types of heart rate**

Heart rate can be found at any point on the body. The pulse under the jaw is called the carotid artery. Pulse at thigh is called femoral artery. The pulse on the inside of your wrist is called radial pulse. The pulse in front of your ankle, arm and under the elbow are called pedal pulse.

As for the doctors, the heart rate is determined by a stethoscope that is place deviated towards the left chest, which is the apex beat measurement (the beat from apex cords) to assess cardiovascular. Apex beat can show us the number, rhythm and the operation of the heart.

#### **Factors influencing heart rate**

Heart rate is influenced by many different factors:

* Environment factor:

When the temperature or the humidity increased, heart pumps a little more blood which increases the heart rate, but usually not more than 5 – 10 beats/min. change of height and the wind can also affect heart rate.

* Nerve system:

If you are having stress, anxiety or suddenly feeling either happy or sad, may make the heart rate increase, which is due to the emotional factor of the brain. And when you exercise, the central nervous system will send impulses through the cardiovascular center in Medulla oblongata to require rapid coordination of both the heart and blood vessels to change blood pressure, increase blood pump to the tissues to meet the body's oxygen need.

* Breathing:

When you breathe in, if you are pay attention enough will notice that your heart beat is slow down, after that it will immediately come back to normal. But patients with congestive lung disease, when they are choking or breathing fast, their heart rate increase to satisfy the body's oxygen need. [16]

* Illness:

Atherosclerosis can lead to coronary heart disease or arrhythmia. Arrhythmia can make the heart beat too fast, too slow or uneven.

Table 2.2: Classify heart rate condition in human

|  |  |
| --- | --- |
| Classes | Heart rate |
| Regular heart rate | 60 beats/min- 100 beats/min |
| Fast heart rate | > 100 beats/min |
| Slow heart rate | < 60 beats/min |

### Blood oxygen levels

#### Blood oxygen levels overview

Oxygen is the most important factor for human life. It is everywhere in the air, when we breath, the oxygen gets into our lungs. Blood and the most important element of the blood is Hemoglobin (Hb) will help transport oxygen from the lungs to necessary places in the body to ensure one's survival. That transportation happens when Hb combines with oxygen create HbO2 (Oxygenated hemoglobin.)

Oxygen saturation in blood is the percentage of Hemoglobin in the blood combined with oxygen. Oxygen saturation in regular person is more than 95%..

#### Types of blood oxygen levels

The concentration of oxygen in the blood depends on the method of measurement, which is divided into two types SpO2 and SaO2. SpO2 is the oxygen saturation in the blood as measured by an oxygen meter based on heart beats. SaO2 is the oxygen saturation in the blood that is measured by measuring arterial blood gas.

While the SaO2 reflects the correct oxygen saturation in blood, the SpO2 has been shown to be inaccurate in detects Hemoglobin diseases due to the absence of abnormal RBCs. SpO2 oxygen saturation is typically about 3% lower than the actual oxygen saturation of SaO2.

#### Factors affect blood oxygen level

SpO2 blood oxygen level is influenced by many things like:

* Carbon monoxide poisoning:

CO is a poisonous gas, produces by burning coal. CO replaces oxygen in the iron-binding site on the Hb molecule, so carbon monoxide poisoning can increase COHb (Hemoglobin with Carbon Monoxide) and reduce HbO2. This will reduce oxygen saturation in the arteries of SaO2. However, SpO2 is higher than SaO2 due to the confusion of the oxygen meter based on the heart beats.

In brief, in the case of carbon monoxide poisoning, SpO2 measured by pulse oximetry does not produce accurate results.

* Anemia:

Anemia means that blood levels of Hemoglobin are lower than normal. in the absence of hypoxemia, the pulse oximetry-based oxygenator is still accurate when the concentration of Hb decreases by 2 g / dL - 3 g / dL. If worst anemia is present when the concentration of Hb decrease by 3g / dL - 9g / dL, SpO2 measurement will be less than SaO2 0.5%.[15]

|  |  |
| --- | --- |
| Classes | Oxygen levels in blood |
| Normal oxygen concentration | 95% |
| Lack of oxygen | < 94 % |

Table 2.3: Classification of oxygen levels in human blood

### Machine learning

There are 2 definitions of Machine Learning are offered. First is described by Arthur Samuel as: "the field of study that gives computers the ability to learn without being explicitly programmed." This is an older, informal definition. Tom Mitchell provides a more modern definition: "A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E."

In general, any machine learning problem can be assigned to one of two broad classifications:

Supervised learning and Unsupervised learning.

There are many different types of machine learning algorithms, with hundreds published each day, and they’re typically grouped by either learning style (i.e. supervised learning, unsupervised learning, semi-supervised learning) or by similarity in form or function (i.e. classification, regression, decision tree, clustering, deep learning, etc.). Regardless of learning style or function, all combinations of machine learning algorithms consist of the following:

* Representation (a set of classifiers or the language that a computer understands)
* Evaluation (aka objective/scoring function)
* Optimization (search method; often the highest-scoring classifier, for example; there are both off-the-shelf and custom optimization methods used)

The fundamental goal of machine learning algorithms is to generalize beyond the training samples i.e. successfully interpret data that it has never ‘seen’ before.

### Support Vector Machine

In machine learning, support vector machines (SVMs, also support vector networks) are supervised learning models with associated learning algorithms that analyze data used for classification and regression analysis. Given a set of training examples, each marked as belonging to one or the other of two categories, an SVM training algorithm builds a model that assigns new examples to one category or the other, making it a non-probabilistic binary linear classifier (although methods such as Platt scaling exist to use SVM in a probabilistic classification setting). An SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall.

In addition to performing linear classification, SVMs can efficiently perform a non-linear classification using what is called the kernel trick, implicitly mapping their inputs into high-dimensional feature spaces.

When data are not labeled, supervised learning is not possible, and an unsupervised learning approach is required, which attempts to find natural clustering of the data to groups, and then map new data to these formed groups. The support vector clustering algorithm created by Hava Siegelmann and Vladimir Vapnik, applies the statistics of support vectors, developed in the support vector machines algorithm, to categorize unlabeled data, and is one of the most widely used clustering algorithms in industrial applications.

More formally, a support vector machine constructs a hyperplane or set of hyperplanes in a high- or infinite-dimensional space, which can be used for classification, regression, or other tasks like outliers detection. Intuitively, a good separation is achieved by the hyperplane that has the largest distance to the nearest training-data point of any class (so-called functional margin), since in general the larger the margin the lower the generalization error of the classifier.

Whereas the original problem may be stated in a finite dimensional space, it often happens that the sets to discriminate are not linearly separable in that space. For this reason, it was proposed that the original finite-dimensional space be mapped into a much higher-dimensional space, presumably making the separation easier in that space. To keep the computational load reasonable, the mappings used by SVM schemes are designed to ensure that dot products may be computed easily in terms of the variables in the original space, by defining them in terms of a kernel function k(x, y) selected to suit the problem. The hyperplanes in the higher-dimensional space are defined as the set of points whose dot product with a vector in that space is constant. The vectors defining the hyperplanes can be chosen to be linear combinations with parameters αi of images of feature vectors xi that occur in the data base. With this choice of a hyperplane, the points x in the feature space that are mapped into the hyperplane are defined by the relation: constant . Note that if k(x, y) becomes small as y grows further away from x, each term in the sum measures the degree of closeness of the test point x to the corresponding data base point xi . In this way, the sum of kernels above can be used to measure the relative nearness of each test point to the data points originating in one or the other of the sets to be discriminated. Note the fact that the set of points x mapped into any hyperplane can be quite convoluted as a result, allowing much more complex discrimination between sets which are not convex at all in the original space.

## Experiamental study

### System overview

*Figure 2‑1: System overview diagram*

The system used Raspberry Pi 3 as central controller to control and get result from plug-and-play modules. After that, it will send data to server. Android app is used for connecting to Raspberry Pi to signal start measuring and get data from server.

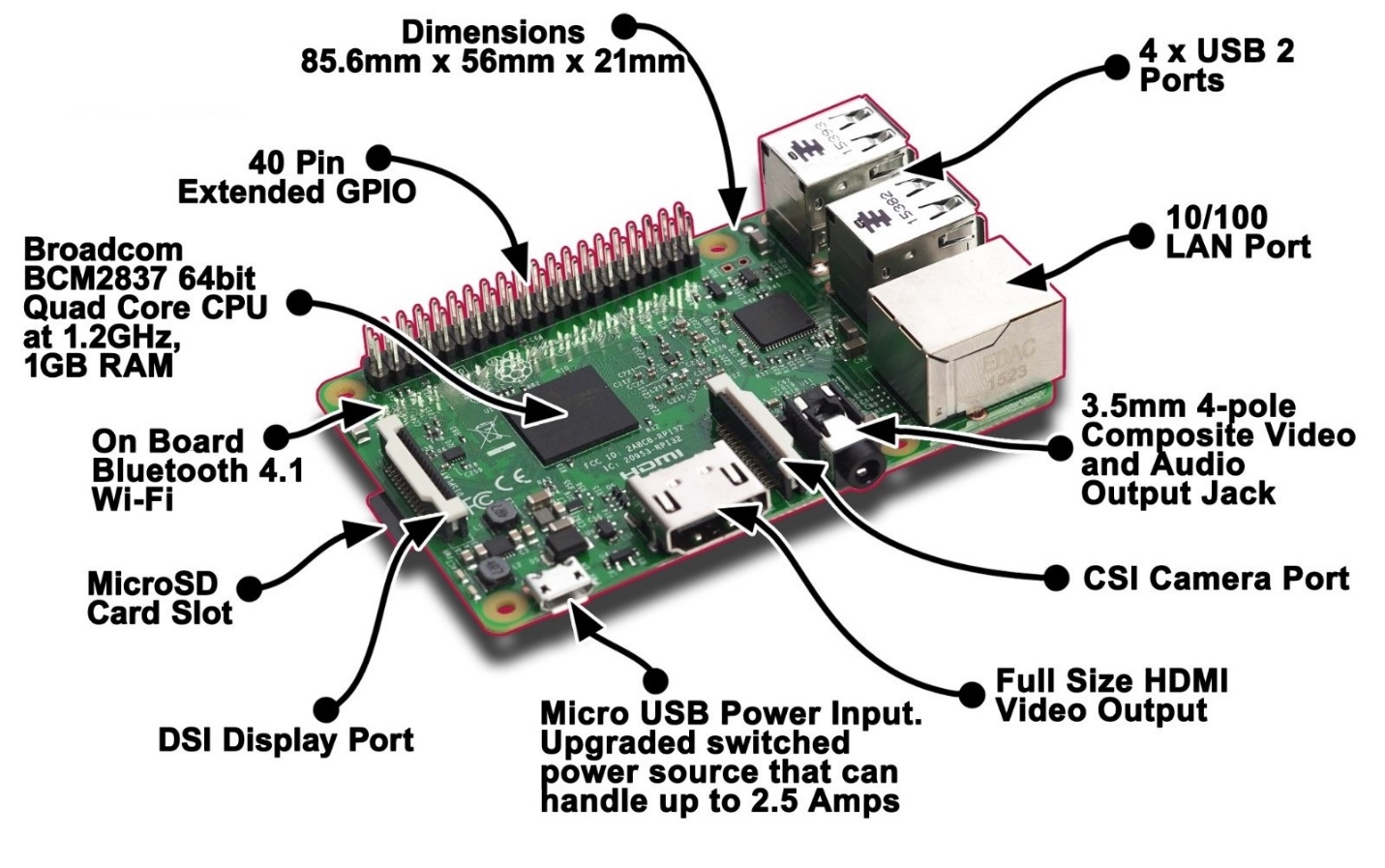
### Devices and tools overview:

Considering how the system works, we came to conclusion that we will use these components for this thesis: Raspberry Pi 3, Arduino Uno, Stm32f103c8t6, temperature sensor, ultra sonic sensor, heart rate and Spo2 sensor.

As for tools, we use Arduino IDE and Keil-C for developing on these components.

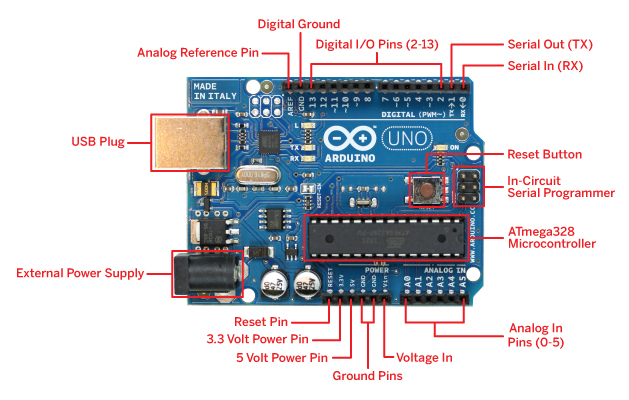
### Study board Raspberry Pi 3

The Raspberry Pi 3 is the third-generation Raspberry Pi. It replaced the Raspberry Pi 2 Model B in February 2016.

* Quad Core 1.2GHz Broadcom BCM2837 64bit CPU
* 1GB RAM
* BCM43438 wireless LAN and Bluetooth Low Energy (BLE) on board
* 40-pin extended GPIO
* 4 USB 2 ports
* 4 Pole stereo output and composite video port
* Full size HDMI
* CSI camera port for connecting a Raspberry Pi camera
* DSI display port for connecting a Raspberry Pi touchscreen display
* Micro SD port for loading your operating system and storing data
* Upgraded switched Micro USB power source up to 2.5A.[3]

*Figure 2‑2 Raspberry Pi 3*

### Arduino Uno overview



|  |  |
| --- | --- |
| Microcontroller | ATmega328 |
| Operating Voltage | 5V DC (through USB port) |
| Clock Speed | 16 MHz |
| Consumed Current | 30mA |
| Input Voltage (recommended) | 7-12V DC |
| Input Voltage (limit) | 6-20V DC |
| Digital I/O Pins | 14 (of which 6 provide PWM output) |
| Analog Input Pins | 6 |
| Max Current on I/O pins | 30 mA |
| Max Output Current (5V) | 500 mA |
| Max Output Current (3.3V) | 50 mA |
| Flash Memory | 32 KB (ATmega328) of which 0.5KB used by bootloader |
| SRAM | 2 KB (ATmega328) |
| EEPROM | 1 KB (ATmega328) |

*Figure 2‑3 Arduino Uno*

Table 2.3 Specifications of Arduino Uno [4]

### Temperature sensor overview:

*MLX90614:*

Melexis MLX90614 is an infrared thermometer designed for non-contact temperature sensing. Using SMBus – an I2C-like interface – to communicate with the chip**.**



*Figure 2‑5 Infrared sensor MLX 90614*

**Some specifications:**

* Factory calibrated
* -40°C to +125°C for sensor temperature
* -70°C to +380°C for object temperature
* ±0.5°C accuracy around room temperatures
* High accuracy of 0.5°C over wide temperature
* 90° Field of view
* 5V version: 4.5 to 5.5V power
* 3V version: 2.6 to 3.6V power
* I2C interface, 0x5A is the fixed 7-bit address. [5]

The special infrared thermopile inside the MLX90614 senses how much infrared energy is being emitted by materials in its field of view, and produces an electrical signal proportional to that. That voltage produced by the thermopile is picked up by the application processor’s 17-bit ADC, then conditioned before being passed over to a microcontroller.

Data receive from measuring temperature of an object is put into register Toreg= 0x07, data receive from measuring temperature of environment is put into register Tareg= 0x06.

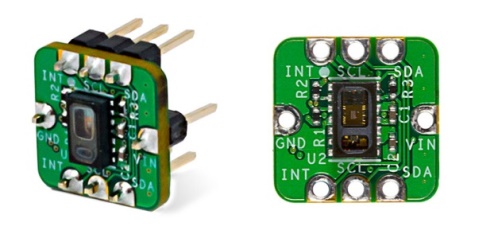
Temperature of object and environment use this equation:

*ToC = (T \*0.02) – 273.15*

Of which T is Toreg or Tareg depends on the measured temperature is of object or environment.

### Heart rate and spo2 sensor overview

#### *MAXREFDES117***:**



***Figure 2‑8*** *Heart-Rate and Pulse-Oximetry Monitor*

Specifications:

* Vin: 2V – 5.5V
* Accuracy:
  + Resting:
    - Spo2: 99%
    - Heart rate: ∓5bpm
  + After workout:
    - Spo2: 92%
    - Heart rate: ∓10bpm [6]

SpO2 calculation is based on the equation shown below. However, determining the constants (C1, C2, and C3) requires a comprehensive clinical study of pulse pulse-oximetry data from a statistically significant population set using this hardware. Such a clinical study is beyond scope of this design. Therefore, the calculated SpO2 value may have can have an error.

SpO2 = C1 × AverageRatio2 + C2 × AverageRatio + C3

Where AverageRatio is the average ratio of IR and red LED readings. C1, C2, and C3 are constants.

### Stm32f103c8t6 overview:

#### **[File:Bluepillpinout.gif](http://wiki.stm32duino.com/images/a/ae/Bluepillpinout.gif)Device overview**

*Figure 2‑9 Stm32f103c8t6*

Specifications:

|  |  |
| --- | --- |
| Microprocessor | STM32F103C8T6 |
| Flash Memory | 64KB |
| RAM | 20KB |
| RTC Crytal | 4-16MHz |
| Peripherals supported | timers, ADC, SPIs, I2 Cs and USARTs |
| Input Voltage | 2.0-3.6V DC |

Table 2‑1 Specification of STM32F103C8 Board

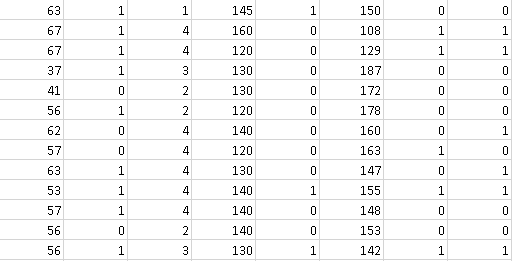
### Heart disease prediction:

#### **Dataset:**

Data for heart disease is from UCI Machine Learning Repository: Heart Disease Data Set. The original dataset consists of 14 attributes but half of them are unusable considered using specific medical equipments to get data for predicition afterward. Because of noises in original data, a pre-process step has to be done before the data can be used for training, data is from databases of 4 hospitals and medical intsitutes. The pre-processing steps are:

* Remove unused attributes from original dataset
* Remove noises from dataset: some values in the dataset is either meaningless or unusable (ex: None, ?, …)

After pre-processing, there are 7 attributes left can be used for training:

* Age
* Sex
* Chest Pain: user has chest pain or not
* Resting Blood Pressure
* Blood Sugar of user
* Heart Rate of user
* Chest pain induced by exercise

*Figure 2‑3 heart disease dataset for training*

### Android app overview

*Figure 2‑4 App overview diagram*

App usage:

* Login screen:
  + Type Email and password then use ‘login’ button to login
  + If don’t have account, tap on ‘sign up’ button to sign up
* Create accout:
  + Required fields:
    - User Name: account name
    - Email: user’s email
    - Password: password for account
* Main menu:
  + Buttons:
    - Connect to measuring system: move to other screen to connect to measuring system
    - User’s Info: move to user’s information screen
    - Health Data: move to user’s health information screen
    - Activity: move to screen shows user’s activities
* Health data:
  + Show measurement results:
    - Heart Rate: heart rate
    - Spo2: oxygen levels in the blood
    - Height: height
    - Weight: weight
    - Temperature: temperature
  + Buttons:
    - Get Health data: get health data
    - Time: choose show data by most recent, week, month
    - Heart: show heart disease probability prediction [17][18]
* Activity:
  + Calories: show burnt calories
  + Walk: show walking time
  + Footstep: show number of footsteps
  + Calories still need: show calories remained need to be burnt in that day. [13][19]
* User Infor:
  + Show basic information of user:
    - First Name
    - Last Name
    - Birthday
    - Age
    - Email
  + Buttons:
    - Save: save changes in this screen
    - Change password: change account password

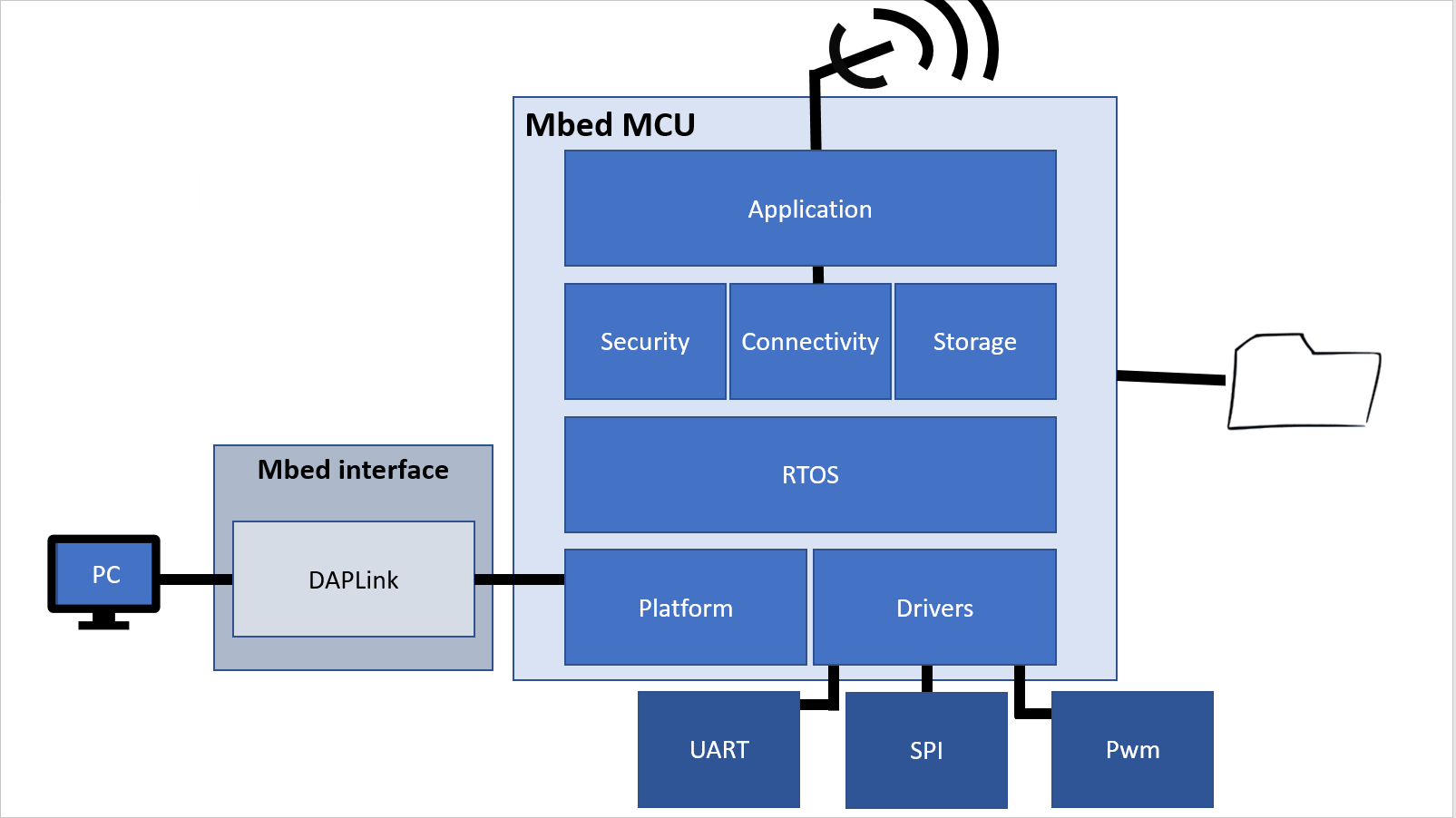
### Operating system

Mbed OS is an open source embedded operating system designed specifically for the devices in the Internet of Things. It includes all the features needed for a connected product based on an Arm Cortex-M microcontroller development.

Mbed OS provides a platform that includes:

* Security foundations.
* Cloud management services.
* Drivers for sensors, I/O devices and connectivity.

Architecture diagram:

The figủe 2-9 below is the basic architecture of an Mbed board:

***Figure 2‑9* Mbed board's hardware architecture**

### Connection between Raspberry Pi and modules

#### **Conection** **between Raspberry Pi and Arduino Uno**

Raspberry Pi connect to Arduino Uno through USB connection. This connection is used for data transmission and providing power to Arduino Uno. This connection used pyserial library on Raspberry Pi. Used functions:

serial.Serial(<serial\_port>, <baud\_rate>, <timeout>): initialize serial connection with serial\_port is the port need to initialize connection, if serial\_port is None then no connection will be established, baud\_rate is set to 9600 for all modules, timeout is for reading data from a connection, it returns immediately when the requested number of bytes are available, otherwise wait until the timeout expires and return all bytes that were received until then.

Serial.isOpen(): for checking if the connection is successful established or not

Serial.write(<data>): Write the bytes data to the port. This should be of type bytes (or compatible such as bytearray or memoryview). Unicode strings must be encoded.

Serial.read(<size>): Read <size> bytes from the serial port. If a timeout is set it may return less characters as requested. With no timeout it will block until the requested number of bytes is read.

Serial.readline(): it only works with a timeout. It depends on having a timeout and interprets that as EOF (end of file). It raises an exception if the port is not opened correctly.

Serial.close(): close the current opening connection.

#### **Conection between Raspberry Pi and** **STM32F103C8T6**

Raspberry Pi connect to Arduino Uno through USB connection. This connection is used for data transmission and providing power to Arduino Uno. This connection used pyserial library on Raspberry Pi and STM32F103C8T6 used Serial API of MBED OS. Functions used on Raspberry Pi is the same as the ones used for connecting with Arduino. Used functions on STM32F103C8T6:

Serial(<PinName tx>, <PinName rx>, <int baud>): Create a Serial port, connected to the specified transmit and receive pins.

Baud(<int baudrate>): set the baud rate of the serial port

format (<bits>, <Parity>, <stop\_bits>): Set the transmission format used by the serial port

gets(<variable>, <size>): get a string with <size> from current connection and put into <variable> for use.

Putc(<char>): send a <char> through serial connection

Printf(<string>): send a <string> through serial connection

### Communicate with oxygen in blood measuring device

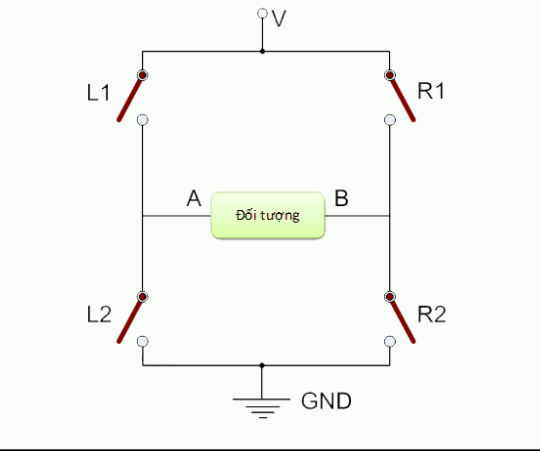
Used library “max30102.h” ( start and get signal from Sensor) and “algorithm.h” (calculate result) provided by manufacturer.

Maxim\_max30102\_read\_reg (): read data from registers.

Maxim\_max30102\_read\_fifo(): read data from fifo (feedback signals of Sensor)

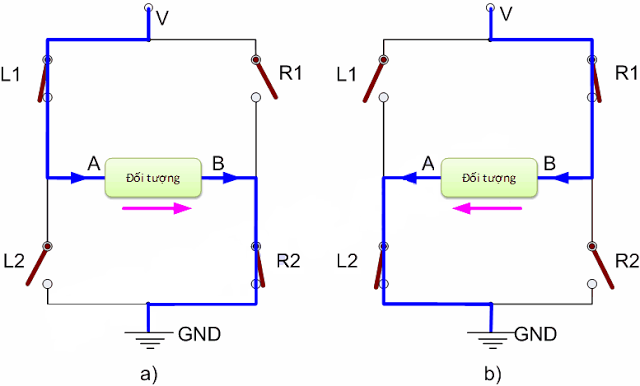
Maxim\_heart\_rate\_and\_satuation():calculate indexes.

### H-Bridge circuit:

H-bridge circuit is used to control the rotation of DC motor via changing direction of the current on motor.

*Figure 2‑10 H-Bridge circuit*

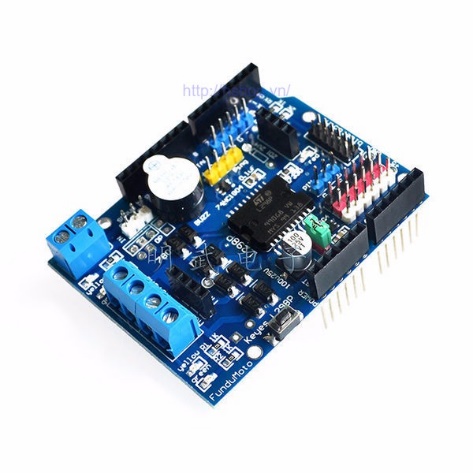
A basic H-circuit consists of a power supply connected to an object (a DC motor) and four switches arranged as Figure 2.3



*Figure 2‑11 Operational principles*

When L1 and R2 are closed (Figure a), the current flows through the object from A to B. Conversely, when R1 and L2 close (Figure b), the current flows through the object from B to A. Thus, All can be adjusted through the object by turning the switches off.

Note: Do not close both L1 and L2 or R switches at the same time! and R2. As such, Vcc and GND will directly connect to each other causing "short circuit" that may cause damage or explosion of equipment.*.*



a) L298 shield b) VNH2SP30

*Figure 2‑12 Driver control Motor*

#### **L298 Shield:**

The L298 Shield is a H hung bridge module for controlling DC motors. The module has plugs directly into the Arduino Uno board. It should be very handy. Motor control over the shield is based on the following pins:

Channel A: Enable : D10 (provide pulse control engine speed).

Direction: D12 (control rotate direction).

Channel B: Enable : D11 (provide pulse control engine speed).

Direction: D13 (control rotate direction).

*Specifications:*

Vin +6.5V ~ +12V

PWRIN: 4.8 – 35V

Iss: <36mA

Maximum current through H-bridge: 2A

Maximum power: 25W (75 oC)

Control voltage: +5V - +7V

Control line: 0 ~ 36mA

#### **VNH2SP30:**

This is the H-bridge circuit with high capacity (30A) suitable for controlling the high capacity Motor replacing L298.

*Specifications:*

Vin +5.5V ~ +16V

Max current: 30A

Continuos current: 14A

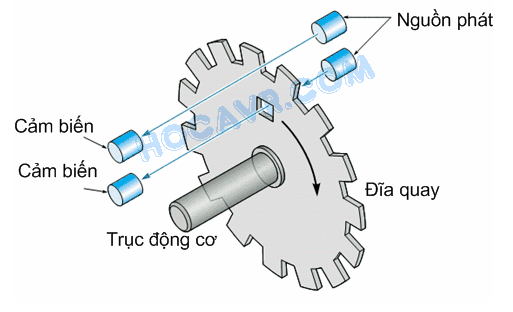
MOSFET resistance: 19m**Ω**

Maximum pulse hash frequency: 20kHz.

Overload and overvoltage protection.

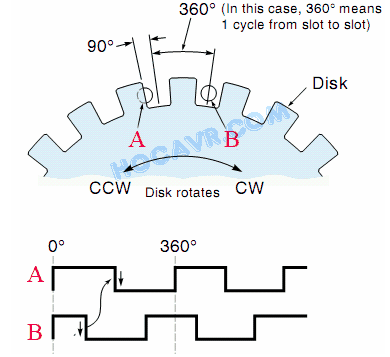
### Encoder:

The encoder is used to count the number of revolutions of the motor and determine the current rotation. From there, it is easy to fill the engine as you like.



*Figure 2‑13 Encoder*

 Encoder usually has 3 channels (3 outputs) including Channel A, Channel B and Channel I (Index). In Figure 2, you notice a small hole in the inside of the spinning disk and a pair of phat-receivers dedicated to the hole. That is the channel I of the encoder. Each time the motor turns a loop, the small hole appears at the location of the transmitter-receiver pair, infrared from the source will penetrate through the small hole to the optical sensor, a signal appearing on the sensor. Thus channel I appears a "pulse" per rotation of the motor. The spinning disc is divided into small grooves and another pair of transducers for these grooves. This is channel A of the encoder, the operation of channel A is similar to channel I, the difference is that during a rotation of the motor, there are N "pulses" appearing on channel A. N is the number of tracks on the disk and is called The resolution of the encoder. Each encoder has different resolutions, sometimes on only a few discs, but there are also thousands of channels. To control the motor, you must know the resolution of the encoder you are using. Resolution affects both control accuracy and control method. Not shown in Figure 2, but on the encoder there is another pair of transponders placed on the same circle with channel A but slightly deviated (M + 0.5 offset), this is channel B of the encoder. The pulse signal from channel B has the same frequency as channel A but the phase difference is 90o. By coordinating channels A and B, the reader knows the direction of the engine. Take a look at the image below [7]



*Figure 2‑14 Channel A and B on Encoder*

Figures 2.7 and 2.8 show the position of the two channel sensors A and B in different phases. When sensor A starts to be covered, sensor B completely receives infrared light through, and vice versa. Low figure is pulse output on two channels. Considering that the motor rotates clockwise, the signal goes from left to right. Take a look at signal A moving from high to low (edge down), channel B is low. Conversely, if the motor turns counterclockwise, the signal "travels" from right to left. At this point, at the down side of channel A, channel B is at a high level. Thus, by combining two channels A and B we not only determine the rotation angle (through the number of pulses) but also know the direction of rotation of the motor (through the level of channel B at the down side of channel A).

### Ultra sonic sensor:

This sensor is used for measuring the distance between the sensor and object.



*Figure 2‑15 Ultra sonic sensor*

The structure of the sensor consists of two speakers 1 for ultrasonic transmission and one for reception of refraction waves. Specifically, from Trig's foot we emit a very short coil and the Echo will emit a High signal as soon as it receives the feedback signal that the Trig foot emits. The sound velocity in the air is 340 m / s => For every 29,412 microseconds the sound travels 1cm. When we get the time the feedback signal can then calculate the difference between the sensor and the object.

*Distance =*

Of which, Distance is the distance to measure, Duration is the time signal feedback.

### USART overview

The term USART stands for "Synchronous and Asynchronous Serial Reveiver and Transmitter," which means synchronous and synchronous serial transmission. [8]

Attention:

The concept of the USART (or UART refers to an asynchronous transmission) typically refers to a hardware device, not just a communication standard.

For example, the RS232 (or COM) standard on personal computers is a combination of UART chips and voltage converters. Signals from the UART chip usually follow the TTL level: the logic level is 5, the low level is 0V. On the other hand, the RS232 signal on the PC is typically -12V for high logic and +12 for low.

USART is fully compatible with UART. There are two changes in the buffer: another buffer is added and the Receiver Shift Register can be used as a third buffer.

Concepts used in this method of communication:

**Baud rate**: transmission speed over a data channel.

**Frame**: The frame consists of bits per transmission.

**Start bit**: start bit is the first bit to be transmitted in a transmission frame, which informs the receiver that a packet is about to be transmitted..

**Data**: data need to be trasnmited.

**Parity bit**: Parity is the bit used to test the correct data transmission

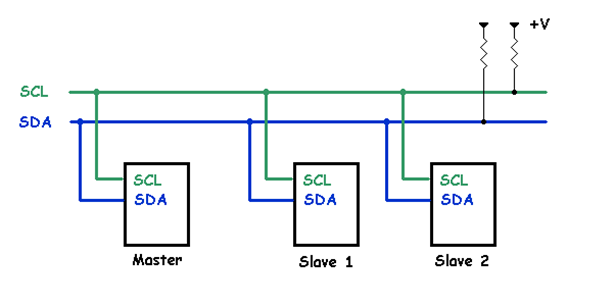
**Stop bits**: Stop bits are one or more bits that tell the receiver that a packet has been sent.

### I2C overview

I2C uses two signal transmission lines:

A single clock (SCL) clock is generated by the master (typically at 100kHz and 400kHz, the highest is 1Mhz and 3.4MHz).

One data path (SDA) in two directions.



*Figure 2‑16 I2C diagram*

There are many devices that can be connected to an I2C bus. However, there is no confusion between devices because each device will be identified by a single address with a single host / I exist for the duration of the connection. Each device can act as a receiving or transmitting device, or it can be transmitted or received. Transmission or receipt depends on whether the device is a master or slave [9].

A device or an IC that connects to an I2C bus, apart from an address (unique) to distinguish, is also configured as a master or slave. Why is this distinctive? This is because on an I2C bus the control is on the host device. The host acts as a clock generator for the whole system, while between host-to-host devices the host device is responsible for clocking and managing the address of the slave device during the delivery. next. The host device plays an active role, while the slave device plays a passive role in communication.

Theoretically and practically I²C uses 7 bits for addressing, so on a bus there can be up to 2 ^ 7 addresses corresponding to 128 devices that can be connected, but only 112 and 16 addresses are available. is used for private purposes. The remaining bits specify whether to read or write data (1 is write, 0 is read)

The strength of I²C is its performance and simplicity: a central control unit can control a whole network of devices with just two control outputs.

### Arduino IDE:

Arduino's integrated development environment (IDE) is a cross-platform application written in Java, and from this IDE it will be used for the Programming programming language and the Wiring project. . It is designed for artists and beginners familiar with the field of software development. It includes a code editor with functions like syntax highlighting, automatic brace matching, and automatic alignment, as well as compile and uploading to the board with just one click. A program or code written for Arduino is called a sketch.

Arduino programs are written in C or C ++. The Arduino IDE comes with a software library called "Wiring", from the original Wiring project, which can make input / output operations easier. The user simply defines two functions to create a cyclic executive program that can be run.

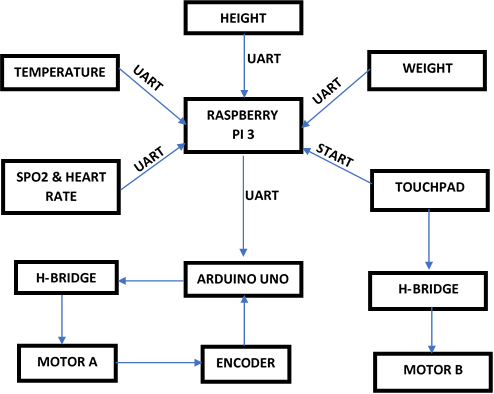
The Arduino IDE uses the GNU toolchain and AVR Libc to compile the program, and uses avrdude to upload the program to the board.

Because the Arduino platform is the Atmel microcontroller, the development environment of Atmel, AVR Studio, or newer Atmel Studio versions can also be used to make software development for Arduino..

# SYSTEM ANALYSIS AND DESIGN

## System general model

### Connection between modules and boards



*Figure 3‑1: Connection diagram*

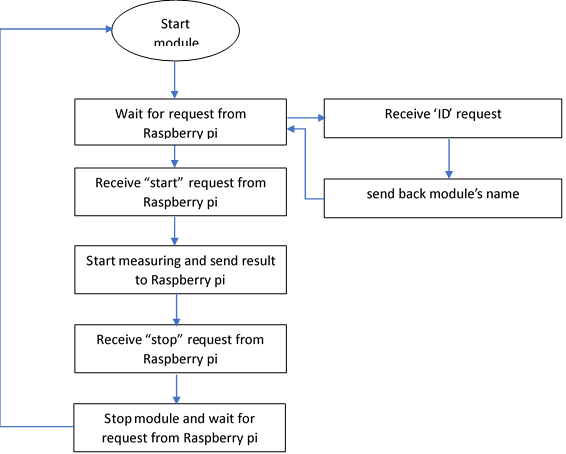
Central controller of the system is Raspberry Pi 3. Modules connect to central controller using USB interface. First, central controller gets result from modules: temperature, height, SPO2, heart rate, scale (weight). After that, it will send the result of height module to Arduino Uno to control Motor A. Touch switch is used for controlling Motor B.

### System operation process:

*Figure 3‑2 Operation process*

The above figure show the working process of the system. After turning on, the system will start checking all the ports of it, searching for connected modules. After that, the system will wait for user’s start signal. After receiving signal, the central controller will send start signal to connected modules to start working at the same time. All the modules will work in parallel and separately. After finishing measuring, the modules will send their results to central controller to and upload to server.

### Module operation process:



*Figure 3‑3 Operation process*

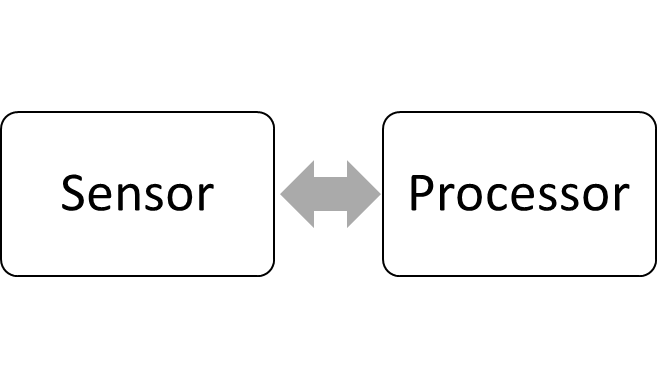
When a module is connected to system, it will start up and wait for request from Raspberry pi. Raspberry pi can send 3 types of request to other modules:

* ID: request name of the module
* Start: request the module to start running
* Stop: request the module to stop running

First, the central controller will send request ‘ID’ to modules with the attempt to identify them. After that, it will send request ‘start’ to signal all identified module to start working. Finally, after getting result, request ‘stop’ will be sent to each module.

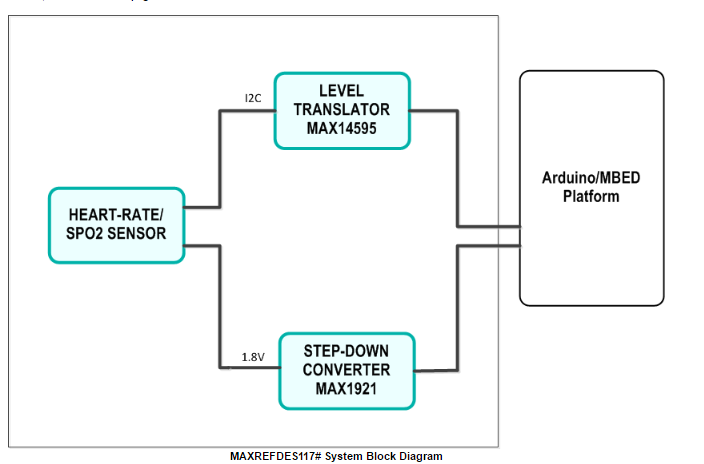
Modules are using STM32F103C8T6 board or Arduino Uno 3 as the processor. Each module include:

* Processor: STM32F103C8T6 board or Arduino Uno 3, responsible for processing data get from sensor before send it to central controller
* Sensor: measuring and feed data to processor to process



*Figure 3‑4 Module general diagram*

### Read data from Heart-Rate and Pulse-Oximetry sensor:



**3**

**2**

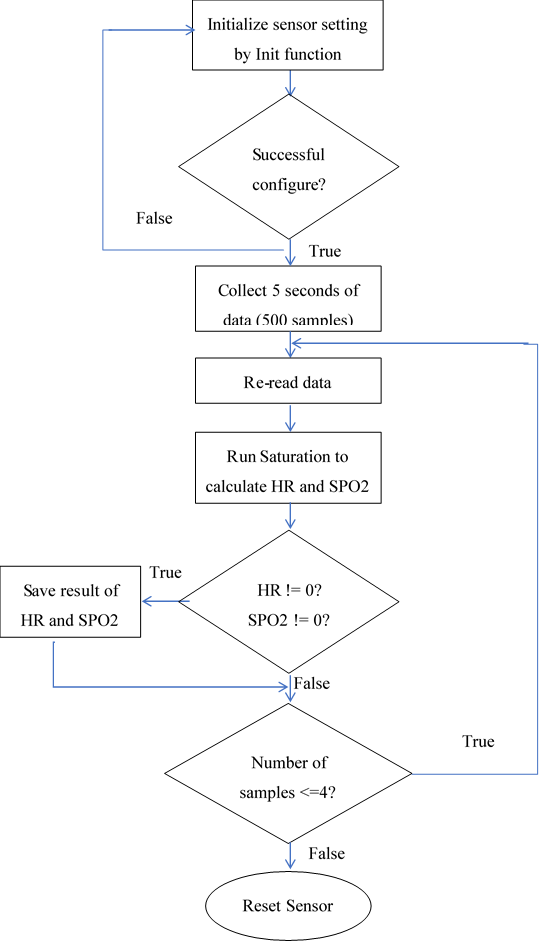
**1**

*Figure 3‑5: Spo2 Sensor Schematic*

Spo2 sensor consists of 3 main blocks:

Maxim 3013/ Heart Rate, Spo2 Sensor: this is the main block for transmitting and receive feedback.

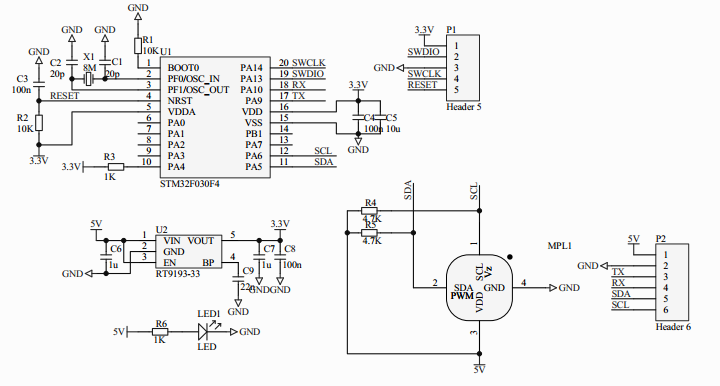
Max 14595/ Lever Traslater: this block is for stabilizing I2C signal.

Max 1921/ StepDown Converter: This block for lowering the voltage from 3.3V or 5V to 1.8V to feed the Sensor.

*Figure 3‑6: Spo2 and HR flowchart*

The figure above shows

### Read data from temperature sensor:



MCU Stm32f10

Bộ khủ nhiễu

MLX 90614

*Figure 3‑7: Schematic Sensor MLX90615*

Device consists of:

Temperature sensor

The MLX90614 reads the signal from the outside via the infrared eye and passes the data using the I2C protocol. Sensors use 2 to 4k7 to stabilize the signal. The input voltage can be 5v or 3.3v.

Noise suppressor.

MCU STM32F10 has been pre-loaded with source code to read the I2C signal from the sensor and output the result. And then the data is translated through the UART protocol.

UART

I2C

STM32F103C8T6

MLX90614

MCU

***Figure 3‑8****: connection from MLX90614 to STM32F103C8T6*

Figure 3-8 shows the connection between MLX90614 and Stm32f103c8t6. MLX90614 and MCU is put on the same board using I2C connection to become sensor MLX90615. From sensor connect to Stm32f10c8t6 using UART connection.

### Android app working process overview

User creates an account with user name, email and password respectively to login

Once created, the account will be moved to the basic information page such as Name, Year of Birth, Career, Daily Targeted Calorie

The main menu selection is connected to the measurement system and performs basic biomimetry measurements such as height, weight, heart rate, oxygen level, and body temperature.

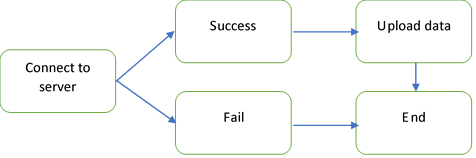
To see the results measured in the health data section, the results are displayed in turn: last measured, last week, last month. Click on Heart Heart button to get predictive results based on the measured results

To test your activity and measure the amount of calories you consume: In the Activity section, the system monitors the current activity of the user and displays the amount of calories consumed, giving a warning of activity if over saturated. long

To change the personal information of the account: In the User Infor section change the information then click save

### Communication between Raspberry Pi and Server:

The figure below shows connection from Raspberry pi to server only established when all the data is ready. When the data is readied, it will call function ‘post(<address>, <data>)’ of library ‘requests’ in python to send data to server.



*Figure 3.8: Data upload to Server diagram*

### Heart disease prediction

The figure below shows the flow of heart disease probability prediction for users.

*Figure 3.9: Heart disease predition flow diagram*

The input in the flow above is a string consists of 7 attributes’ values, which attributes are used for training.

# TEST RESULTS AND EVALUATION

## Test process

### Components for testing

Components for testing:

Raspberry Pi 3.

Arduino Uno.

Spo2 sensor.

Temperature sensor.

Ulrtra sonic sensor.

Sheild L298.

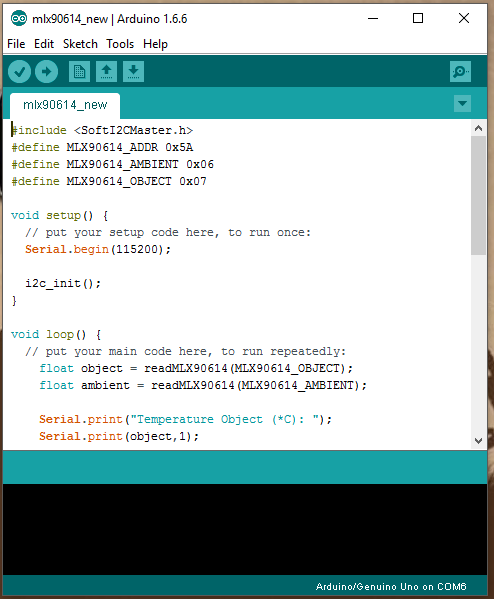
Kit VNH2SP30.

Motor DC.

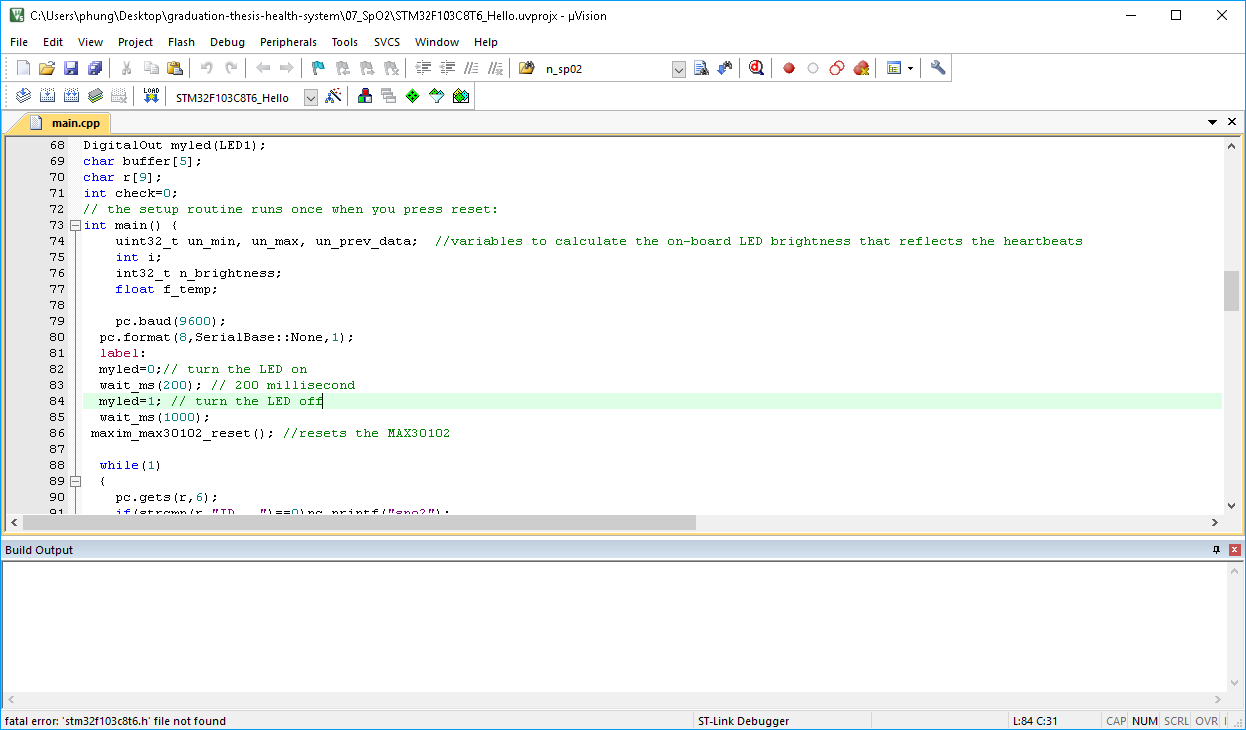
Stm32f103c8t6

UART – USB converter cp2102

Raspberry pi 3 is the central board, Arduino Uno and Stm32f103c8t6 are for plug-and-play controlling

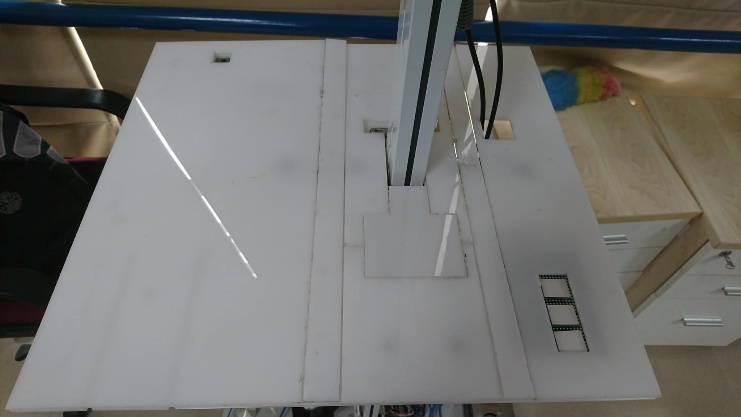


*Figure 4-1: Download code into Arduino Uno*



*Figure 4-2: Download code into Stm32f103c8t6*

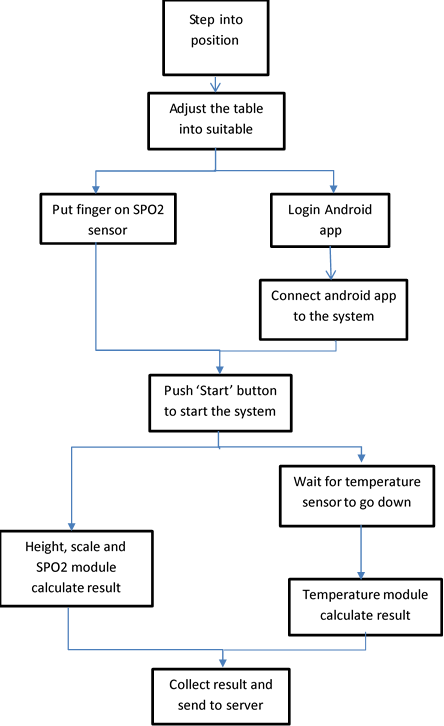




*Figure 4-3: Components after connected*

*Figure 4-3: Components after connected*

### Steps to use the system:



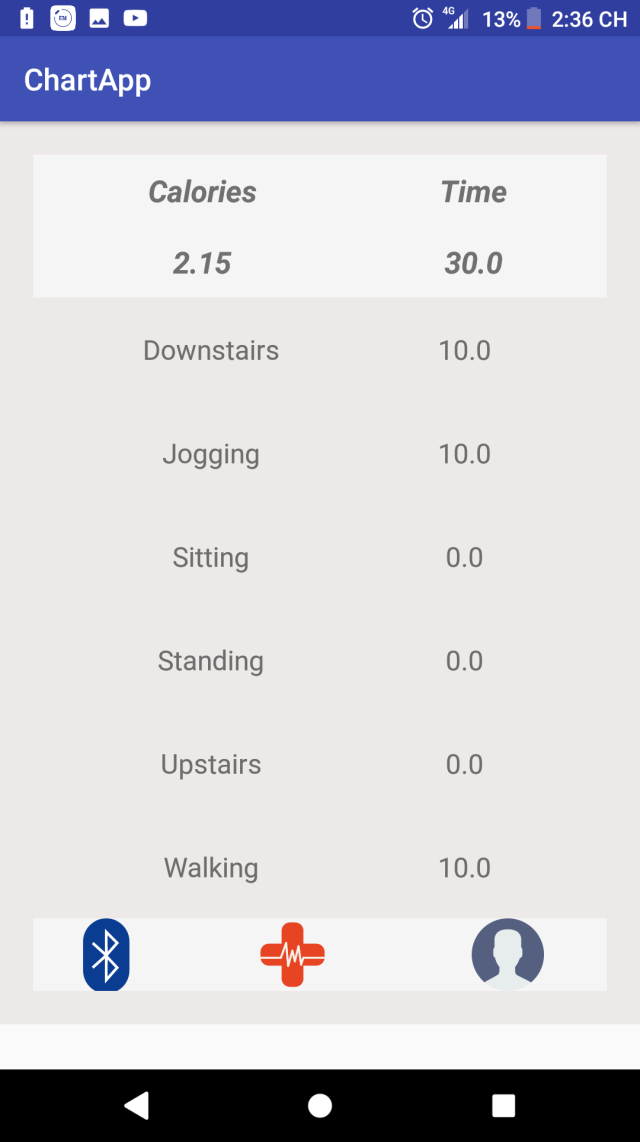
***Figure 4-4****: system process diagram*

The figure above shows how to use the system. First, step into the right position. Then put the index finger on Spo2 sensor, login on android app, connect to the system. To start the system, push start button on android app. After that, all the modules will start working at the same time. After the results are collected, they will be sent to server.

### Result:

#### C:\Users\admin\Downloads\26972537_1149000808536983_801870078_o.png**Android application result**

Result screen of android app:



*Figure 4-5: android app result screen*

The figure above shows the result screen of android app, the left image is the main result screen, the right image is the activity result screen. In the main result screen, data shows on the screen is the latest data of the loged in account on server. In the activity result screen, user’s active time of the listed activities and calculate burnt calories based on them.

#### **Temperature result:**

Table 4.1 Temperature result

|  |  |  |
| --- | --- | --- |
| thermometer | sensor | error |
| 36.51 | 35.91 | 0.6 |
| 36.6 | 35.66 | 0.94 |
| 37.27 | 36.13 | 1.14 |
| 36.93 | 36.21 | 0.72 |
| 37.16 | 36.11 | 1.05 |
| 36.67 | 36.11 | 0.56 |
| 36.78 | 35.84 | 0.94 |
| 37.48 | 36.32 | 1.16 |
| 36.62 | 35.47 | 1.15 |
| 36.99 | 36.32 | 0.67 |
| 37.17 | 35.99 | 1.18 |
| 37.29 | 36.62 | 0.67 |
| 37.3 | 36.48 | 0.82 |
| 36.72 | 36.11 | 0.61 |
| 37.39 | 36.84 | 0.55 |
| 36.59 | 35.85 | 0.74 |
| 37.17 | 36.4 | 0.77 |
| 36.52 | 35.96 | 0.56 |
| 37.4 | 36.87 | 0.53 |
| 37.49 | 36.71 | 0.78 |
| 36.78 | 35.99 | 0.79 |
| 36.92 | 36.12 | 0.8 |
| 36.6 | 35.4 | 1.2 |
| 37.39 | 36.33 | 1.06 |
| 37.19 | 36.59 | 0.6 |
| 36.74 | 35.98 | 0.76 |
| 37.5 | 36.73 | 0.77 |
| 36.63 | 35.86 | 0.77 |
| 36.59 | 35.99 | 0.6 |
| 36.68 | 36.11 | 0.57 |
| 37.31 | 36.66 | 0.65 |
| 37.34 | 36.56 | 0.78 |
| 37.42 | 36.82 | 0.6 |
| 36.66 | 35.54 | 1.12 |
| 37.4 | 36.5 | 0.9 |
| 36.69 | 35.76 | 0.93 |
| 36.65 | 35.94 | 0.71 |
| 37 | 36.41 | 0.59 |
| 36.87 | 36.35 | 0.52 |
| 37.06 | 36.33 | 0.73 |
| 36.77 | 36.24 | 0.53 |
| 37.02 | 35.97 | 1.05 |
| 37.48 | 36.73 | 0.75 |
| 37.39 | 36.72 | 0.67 |
| 37.26 | 36.25 | 1.01 |
| 37.11 | 36.26 | 0.85 |
| 36.52 | 35.86 | 0.66 |
| 36.95 | 35.81 | 1.14 |

##### Evaluation:

The test was done with conditions:

* The tester’s fore head is 1.5cm away from the sensor
* Tested on 2 devices: thermometer and sensor
* Tested on different environments
* Tested 48 times

Result:

* Thermometer: temperature ranges from 36.5 oC to 37.46 oC
* Sensor: temperature ranges from 34.67 oC to 35.73 oC
* Average error of 2 methods: 0.8 oC
* Standard deviation: 0.37
* The error of the current system compare to real value ranges from 1.4% to 3.3%
* The error of the previous system compare to real value ranges from 4.2% to 5.6%

→ current system has error percentage near the expected range (± 5%) than previous system

From the results of the survey, the temperature measured by a mercury thermometer is higher than the infrared sensor. The reason is that the infrared sensor temperature is measured only as a peripheral body temperature while the temperature of the mercury thermometer is measured as the central body temperature. In order to overcome this difference, we can obtain the results obtained by the external infrared sensor plus the mean error between the two methods is approximately 1.75oC.

#### **Spo2 result:**

Table 4.2 Spo2 result

|  |  |  |
| --- | --- | --- |
| Spo2 meter | Heat rate/Spo2 sensor | error |
| 98 | 97 | 1 |
| 98 | 98 | 0 |
| 98 | 97 | 1 |
| 98 | 97 | 1 |
| 99 | 99 | 0 |
| 97 | 97 | 0 |
| 97 | 94 | 3 |
| 97 | 96 | 1 |
| 98 | 96 | 2 |
| 99 | 97 | 2 |
| 99 | 99 | 0 |
| 98 | 98 | 0 |
| 97 | 96 | 1 |
| 98 | 97 | 1 |
| 97 | 97 | 0 |
| 97 | 96 | 1 |
| 98 | 97 | 1 |
| 98 | 97 | 1 |
| 98 | 96 | 2 |
| 98 | 97 | 1 |
| 98 | 94 | 4 |
| 98 | 96 | 2 |
| 97 | 96 | 1 |
| 97 | 97 | 0 |
| 98 | 97 | 1 |
| 98 | 97 | 1 |
| 97 | 97 | 0 |
| 99 | 98 | 1 |
| 97 | 96 | 1 |
| 98 | 98 | 0 |
| 98 | 98 | 0 |
| 99 | 98 | 0 |
| 97 | 96 | 1 |
| 98 | 98 | 0 |
| 98 | 97 | 1 |
| 98 | 97 | 0 |
| 98 | 96 | 2 |
| 99 | 99 | 0 |
| 98 | 96 | 2 |
| 98 | 98 | 0 |
| 99 | 98 | 1 |
| 98 | 96 | 2 |
| 99 | 98 | 1 |
| 99 | 96 | 3 |
| 99 | 99 | 0 |
| 99 | 98 | 1 |
| 99 | 98 | 1 |
| 98 | 96 | 2 |

##### Evaluation:

The test was conducted to measure oxygen levels in blood from spo2 meter and sensor at the left index finger:

* Spo2 meter: result ranges from 97% to 99%
* Sensor: result ranges from 94% to 99%
* Average error is 0.98%
* Standard deviation is 1.13
* The error of the current system compare to real value ranges from 0% to 4.08%
* The error of the previous system compare to real value ranges from 0% to 12.2%

→ current system has higher accuracy than the previous system

From the result, the difference from sensor and spo2 meter is not so much.

#### **Heart rate result**

Table 4.3 Heart rate result

|  |  |  |
| --- | --- | --- |
| Omron meter | Heart rate/spo2 sensor | Error |
| 86 | 81 | 5 |
| 80 | 80 | 0 |
| 75 | 70 | 5 |
| 79 | 79 | 0 |
| 79 | 79 | 0 |
| 69 | 69 | 0 |
| 75 | 74 | 1 |
| 86 | 80 | 6 |
| 80 | 80 | 0 |
| 75 | 75 | 0 |
| 70 | 70 | 0 |
| 78 | 78 | 0 |
| 82 | 82 | 0 |
| 87 | 80 | 7 |
| 77 | 76 | 1 |
| 75 | 75 | 0 |
| 77 | 77 | 0 |
| 80 | 80 | 0 |
| 80 | 80 | 0 |
| 68 | 68 | 0 |
| 77 | 77 | 0 |
| 70 | 71 | 1 |
| 77 | 75 | 2 |
| 69 | 73 | 4 |
| 72 | 72 | 0 |
| 83 | 82 | 1 |
| 74 | 74 | 0 |
| 85 | 78 | 7 |
| 85 | 85 | 0 |
| 71 | 70 | 1 |
| 77 | 77 | 0 |
| 81 | 81 | 0 |
| 86 | 83 | 3 |
| 79 | 82 | 3 |
| 82 | 82 | 0 |
| 72 | 70 | 2 |
| 83 | 88 | 5 |
| 76 | 77 | 1 |
| 69 | 69 | 0 |
| 72 | 72 | 0 |
| 79 | 78 | 1 |
| 82 | 82 | 0 |
| 71 | 73 | 2 |
| 70 | 69 | 1 |
| 83 | 83 | 0 |
| 70 | 70 | 0 |
| 87 | 87 | 0 |
| 73 | 73 | 0 |

##### Evaluation:

The test was conducted on sensor and Omron device.

* Omron result: ranges from 68 bpm to 87 bpm
* Sensor result: ranges from 68 bpm to 88 bpm
* Average error: 1.23 bpm
* Standard deviation: 5.18
* The error of the current system compare to real value ranges from -6% to 8.2%
* The error of the previous system compare to real value ranges from 0% to 8.7%

→ current system has lower accuracy than the previous system due to different in devices

From the result, the difference between sensor and Omron device is about 1.23, this is not a big difference.

#### **Weight result**

Table 4.4 weight result

|  |  |  |
| --- | --- | --- |
| scale | sensor | error |
| 69.1 | 69 | 0.1 |
| 70 | 70.7 | 0.7 |
| 68.9 | 69.4 | 0.5 |
| 69.8 | 70.9 | 1.1 |
| 68.9 | 69.3 | 0.4 |
| 69.7 | 69.5 | 0.2 |
| 70.5 | 70.1 | 0.4 |
| 70.5 | 69.2 | 1.3 |
| 68.6 | 70.4 | 1.8 |
| 70.9 | 70.5 | 0.4 |
| 68.5 | 69.4 | 0.9 |
| 69.5 | 69.1 | 0.4 |
| 70.6 | 70.2 | 0.4 |
| 69.7 | 69.6 | 0.1 |
| 68.5 | 69.2 | 0.7 |
| 69.7 | 69.6 | 0.1 |
| 71 | 70.9 | 0.1 |
| 68.6 | 70.6 | 2 |
| 69 | 69.9 | 0.9 |
| 68.7 | 69.6 | 0.9 |
| 70.2 | 70.6 | 0.4 |
| 70.5 | 70.9 | 0.4 |
| 70.4 | 69.3 | 1.1 |
| 69.6 | 69.8 | 0.2 |
| 68.9 | 70.4 | 1.5 |
| 69.7 | 69.6 | 0.1 |
| 70.2 | 71 | 0.8 |
| 69.2 | 69.1 | 0.1 |
| 68.8 | 69.9 | 1.1 |
| 70.7 | 69.4 | 1.3 |
| 69.7 | 70.2 | 0.5 |
| 70.9 | 71 | 0.1 |
| 68.5 | 69 | 0.5 |
| 70.1 | 70.5 | 0.4 |
| 70.2 | 70.7 | 0.5 |
| 69.5 | 69.7 | 0.2 |
| 69.4 | 69.3 | 0.1 |
| 69 | 69 | 0 |
| 70.3 | 69.4 | 0.9 |
| 70.4 | 69 | 1.4 |
| 68.8 | 69.2 | 0.4 |
| 68.6 | 69.6 | 1 |
| 71 | 69 | 2 |
| 69 | 69.2 | 0.2 |
| 70.6 | 70.7 | 0.1 |
| 69.1 | 69.9 | 0.8 |
| 69.7 | 70.2 | 0.5 |
| 70.6 | 70.6 | 0 |

##### Evaluation :

* Scale: result ranges from 68.5 kgs to 71 kgs
* Sensor: ranges from 69 kgs to 71 kgs
* Average error: 0.63 kgs
* Standard deviation: 0.66
* The error of the system compare to real value is ±2.9%
* The accuracy is still in expected range (± 5%)
* The previous system did not have this module

From the result, the difference is low so the sensor is quite accurate compare to the scale.

#### **Height result**

Table 4.5: Height result

|  |  |  |
| --- | --- | --- |
| original height | sensor | error |
| 180 | 181 | 1 |
| 180 | 179 | 1 |
| 180 | 180 | 0 |
| 180 | 180 | 0 |
| 180 | 181 | 1 |
| 180 | 180 | 0 |
| 180 | 181 | 1 |
| 180 | 179 | 1 |
| 180 | 180 | 0 |
| 180 | 180 | 0 |
| 180 | 180 | 0 |
| 180 | 180 | 0 |
| 180 | 181 | 1 |
| 180 | 179 | 1 |
| 180 | 179 | 1 |
| 180 | 181 | 1 |
| 180 | 180 | 0 |
| 180 | 181 | 1 |
| 180 | 181 | 1 |
| 180 | 180 | 0 |
| 180 | 180 | 0 |
| 180 | 179 | 1 |
| 180 | 180 | 0 |
| 180 | 180 | 0 |
| 180 | 179 | 1 |
| 180 | 181 | 1 |
| 180 | 181 | 1 |
| 180 | 181 | 1 |
| 175 | 176 | 1 |
| 175 | 175 | 0 |
| 175 | 175 | 0 |
| 175 | 175 | 0 |
| 175 | 174 | 1 |
| 175 | 174 | 1 |
| 175 | 176 | 1 |
| 175 | 174 | 1 |
| 175 | 174 | 1 |
| 175 | 176 | 1 |
| 175 | 174 | 1 |
| 175 | 175 | 0 |
| 175 | 174 | 1 |
| 175 | 174 | 1 |
| 175 | 176 | 1 |
| 175 | 175 | 0 |
| 175 | 176 | 1 |
| 175 | 176 | 1 |
| 175 | 174 | 1 |
| 175 | 175 | 0 |

##### Evaluation :

Original height:180 cm and 175 cm

Sensor: ranges from:

* + 179 cm to 181 cm for original height 180 cm
  + 174 cm to 176 cm for original height 175 cm

Average error: 0.625 cm

Standard deviation: 2.73

The error of the system compare to real value is ±0.6%

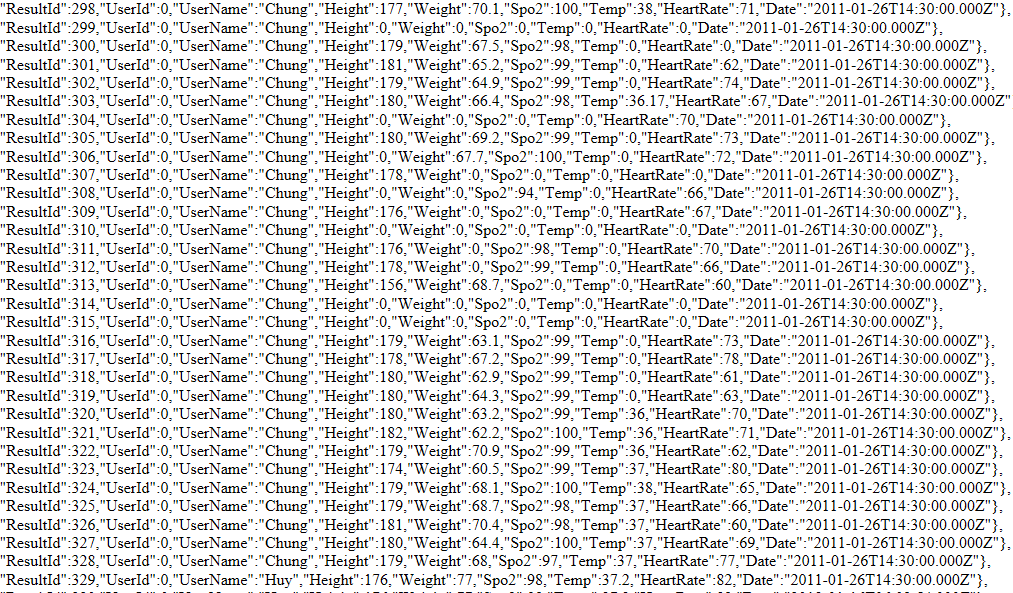
The accuracy is high enough for using in reality

The previous system did not have this module

From the result, the difference is low so the sensor is accurate compare to the original height of the testers.

#### **Result from server**

#### Table 4.6: Result from Server



Base on result from test, we have the conclusion:

* The height and weight result from sensor are pretty much high accurate
* The result from spo2 / heart rate, temperature sensor still have high difference compare to the devices on market.

# CONCLUSION AND REQUEST

## Conclusion:

Throughout the study period, the team gained a lot of valuable knowledge as well as learned the skills needed to complete their work such as:

- Read a Schematic of a device.

- Consolidate and enhance programming knowledge of devices based on Arduino IDE and Keil C.

- Connect and communicate between Raspberry Pi 3 and Arduino Uno.

- Connect and communicate between Raspberry Pi 3 with Stm32f103c8t6

- Connection between Raspberry Pi 3 and Server.

- The ability to collect and analyze documents.

- Teamwork skill.

- Soldering skills.

- Skills to write a scientific research report

### Achievements:

The results of the group after the implementation of the project:

- Build complete system with plug-and-play modules for: Spo2, Temperature and Heart Rate.

- Run the system successfully and take blood oxygen levels, heart rate, temperature, height and weight from plug-and-play modules.

- The system works quite well.

- Build Cloud Data Receiving and Sending System.

- The accuracy of Spo2, Temperature is higher than the previous system.

### Failure:

Due to limited research time as well as budget for implementation, there are some limitations as follows:

- When execute wrong step while using, error may occur in some modules.

- In some rare cases, errors can occur. The most is the heart rate value.

- Due to limited mechanical capabilities, the equipment is still not completely stable.

- There are still some unreasonable things in the design of the system.

- Due to using non-invation method, many vital signs cannot be measured will affect the accucracy of prediction in some situations

- Android application user interface still has some inconveniences for users.

## Development in the future

Continue to improve the system and expand the scale, connecting with other health monitoring devices.

- need to improve the exterior design

- complete application, data management and storage system on PC / phone.

- Integration with other automation systems (electrochemical, breathing, ...) to form a comprehensive Smart Heathy. Equipment needed for SmartHome through Internet of Things.

- Summary of measurement results, when the abnormal index automatically send the measurement results to the doctor through the Internet connection

- Connect to peripherals such as printers / faxes to get results directly on paper.

- increase the number of diagnosable diseases

**REFERENCE**

[1]"Tỉ lệ tử vong do bệnh không lây nhiễm ở Việt Nam cao hơn thế giới", *Moh.gov.vn*, 2018. [Online]. Available: <http://moh.gov.vn/news/pages/tinkhac.aspx?ItemID=2736>

[2] Wikipedia*, “Vital signs”*. [Online]. Available: <https://en.wikipedia.org/wiki/Vital_signs>

[3] Raspberry Pi, *raspberrypi.org* . [Online]. Available: <https://www.raspberrypi.org/products/raspberry-pi-3-model-b/>

[4] Arduino Software (IDE), *arduino.cc*. [Online]. Available: <https://www.arduino.cc/en/guide/environment>

[5]  Melenis90615, *melexis.com*. [Online]. **Available:** https://[www.melexis.com/en/product/mlx90615/digital-plug-play-infrared-thermometer-ultra-small-to-can](https://www.melexis.com/en/product/mlx90615/digital-plug-play-infrared-thermometer-ultra-small-to-can)

[6] MAXREFDES117, *maximintegrated.com*. [Online]. **Available:** <https://www.maximintegrated.com/en/design/reference-design-center/system-board/6300.html>

[7] Encoder, *hocavr.com*. [Online]. Available: <http://www.hocavr.com/index.php/app/dcservo>

**[8] UART,** ***whatis.techtarget.com*.** [Online]. **Available:** <http://whatis.techtarget.com/definition/UART-Universal-Asynchronous-Receiver-Transmitter>

[9] I2C, *arduino.vn*. [Online]. **Available:** <http://arduino.vn/bai-viet/1053-giao-tiep-i2c-voi-nhieu-module>

[10]"US20070156456A1 - System for Monitoring Healthcare Related Activity In A Healthcare Enterprise - Google Patents", Patents.google.com, 2018. [Online]. Available: <https://patents.google.com/patent/US20070156456A1/en>

[11]M. Albarrak, K. Banno, A. Sabbagh, K. Delaive, R. Walld, J. Manfreda and M. Kryger, "Utilization of Healthcare Resources in Obstructive Sleep Apnea Syndrome: a 5-Year Follow-Up Study in Men Using CPAP", 2018 .

[12] Who.int, 2018. [Online]. Available: <http://www.who.int/nmh/countries/vnm_en.pdf>

[13]"Calories Burned by Sitting - light office work, in general", Calorielab.com, 2018. [Online]. Available: <http://calorielab.com/burned/?mo=ac&ac=21005&ti=Sitting+-+light+office+work%2C+in+general&q=&wt=150&un=lb&kg=68>

[14]S. Buddies and S. Buddies, "Sweaty Science: How Does Heart Rate Change with Exercise?", *Scientific American*, 2018. [Online]. Available: <https://www.scientificamerican.com/article/bring-science-home-heart-rate-excercise/>

[15]G. Basaranoglu, M. Bakan, T. Umutoglu, S. Zengin, K. Idin and Z. Salihoglu, "Comparison of SpO2 values from different fingers of the hands", 2018

[16]D. Oh, H. Hong and B. Lee, "The effects of strenuous exercises on resting heart rate, blood pressure, and maximal oxygen uptake", 2018

[17]D. Freedman, L. Khan, W. Dietz, S. Srinivasan and G. Berenson, "Relationship of Childhood Obesity to Coronary Heart Disease Risk Factors in Adulthood: The Bogalusa Heart Study", 2018

[18]R. Bhopal, N. Unwin, M. White, J. Yallop, L. Walker, K. Alberti, J. Harland, S. Patel, N. Ahmad, C. Turner, B. Watson, D. Kaur, A. Kulkarni, M. Laker and A. Tavridou, "Heterogeneity of coronary heart disease risk factors in Indian, Pakistani, Bangladeshi, and European origin populations: cross sectional study".

[19]"Patent US6013009 - Walking/running heart rate monitoring system", Google Books, 2018. [Online]. Available: <https://www.google.com/patents/US6013009>

[20]A. Kuskowska-Wolk, P. Karlsson, M. Stolt and S. Rössner, "The predictive validity of body mass index based on self-reported weight and height.", Europepmc.org, 2018. [Online]. Available: <http://europepmc.org/abstract/med/2793299>